CLAIMS

1. A method in a computer system for managing a plurality of types of enterprise portfolio data, the portfolio data having investment items, the computer system defining an object type to correspond to each of the plurality of types of portfolio data, each object type defining a plurality of attributes, comprising:

for each investment item,

instantiating an object as an instance of the object type that corresponds to the investment item, the instantiated object containing a plurality of attributes; and

setting values of the plurality of attributes of the instantiated object based upon data associated with the investment item,

wherein at least one of the investment items is a financial investment item that reflects financial investment data and at least another one of the investment items is a project management item that reflects project related data and the instantiated objects that correspond to the financial investment item and the project management item are managed as interchangeable items.

2. The method of claim 1, further comprising:

receiving additional portfolio data;

adding a new object type to correspond to the received data, the new object type defining a new plurality of attributes; and

instantiating a new object as an instance of the new object type to correspond to the received data and setting values of the plurality of attributes defined by the new object type based upon the received data.

3. The method of claim 1 wherein the sources for the portfolio data reside in a plurality of separate data repositories that belong to distinct organizations within an enterprise.

4. The method of claim 1 wherein the investment items are at least two of financial investments, project management, collections of projects, products, programs, assets, human resources, portfolios, initiatives, applications, operations, processes, and activities.

5. The method of claim 1, the enterprise portfolio data comprising at least two of engineering, marketing, product management, manufacturing, sales, information technology, finance, human resources, research, development, and professional services portfolios.

6. The method of claim 1 wherein at least one of the attributes has multiple dimensions yielding multiple sets of values for the at least one attribute.

7. The method of claim 6 wherein the attribute dimensions are at least two of actual value, baseline value, plan value, target value, and scenario value.

8. A computer-readable memory medium containing instructions to control a computer processor to manage a plurality of types of enterprise portfolio data, the portfolio data having investment items, the computer system defining an object type to correspond to each of the plurality of types of portfolio data, each object type defining a plurality of attributes, by:

for each investment item,

instantiating an object as an instance of the object type that corresponds to the investment item, the instantiated object containing a plurality of attributes; and

setting values of the plurality of attributes of the instantiated object based upon data associated with the investment item,

wherein at least one of the investment items is a financial investment item that reflects financial investment data and at least another one of the investment items is a project management item that reflects project related data and the instantiated objects that correspond to the financial investment item and the project management item are managed as interchangeable items.

9. The memory medium of claim 8, further comprising instructions that control the computer processor by:

receiving additional portfolio data;

adding a new object type to correspond to the received data, the new object type defining a new plurality of attributes; and

instantiating a new object as an instance of the new object type to correspond to the received data and setting values of the plurality of attributes defined by the new object type based upon the received data.

10. The memory medium of claim 8 wherein the sources for the portfolio data reside in a plurality of separate data repositories that belong to distinct organizations within an enterprise.

11. The memory medium of claim 8 wherein the investment items are at least two of financial investments, project management, collections of projects, products, programs, assets, human resources, portfolios, initiatives, applications, operations, processes, and activities.

12. The memory medium of claim 8, the enterprise portfolio data comprising at least two of engineering, marketing, product management, manufacturing, sales, information technology, finance, human resources, research, development, and professional services portfolios.

13. The memory medium of claim 8 wherein at least one of the attributes has multiple dimensions yielding multiple sets of values for the at least one attribute.

14. The memory medium of claim 13 wherein the attribute dimensions are at least two of actual value, baseline value, plan value, target value, and scenario value.

15. An portfolio management system for managing portfolios of investment items, comprising:

an object type management system that defines types of investments for which objects can be instantiated and a plurality of attributes that are associated with each type;

an object instantiation system that is structured to:

instantiate an object as an instance of the object type that corresponds to an investment item, the instantiated object containing a plurality of attributes; and

set values of the plurality of attributes of the instantiated object based upon data associated with the investment item,

wherein at least one of the investment items is a financial investment item that reflects financial investment data and at least another one of the investment items is a project management item that reflects project related data and the instantiated objects that correspond to the financial investment item and the project management item are managed as interchangeable items.

16. The system of claim 15 wherein the portfolio management system is an enterprise portfolio management system.

17. The system of claim 15 wherein

the object type management system is further structured to:

add a new object type to correspond to a received additional portfolio investment item, the new object type defining a new plurality of attributes; and

the object instantiation system is further structured to:

instantiate a new object as an instance of the new object type to correspond to the received investment item; and

set values of the plurality of attributes defined by the new object type based upon the received investment item.

18. The system of claim 15 wherein the sources for the portfolio investment items reside in a plurality of separate data repositories that belong to distinct organizations within an enterprise.

19. The system of claim 15 wherein the investment items are at least two of financial investments, project management, collections of projects, products, programs, assets, human resources, portfolios, initiatives, applications, operations, processes, and activities.

20. The system of claim 15, the portfolio comprising at least two of engineering, marketing, product management, manufacturing, sales, information technology, finance, human resources, research, development, and professional services portfolios.

21. The system of claim 15 wherein at least one of the attributes has multiple dimensions yielding multiple sets of values for the at least one attribute.

22. The system of claim 21 wherein the attribute dimensions are at least two of actual value, baseline value, plan value, target value, and scenario value.

23. A computer system for managing and analyzing enterprise portfolio data, comprising:

a portfolio manager component that is structured to add to a portfolio representation structure items that correspond to transactions on the portfolio data; and

a portfolio analyzer component that is structured to present a plurality of views of the portfolio data as represented by the portfolio representation structure, wherein the views dynamically calculate and present multi-dimensional characterizations of the portfolio data while items are added using the portfolio manager.

24. The system of claim 23 wherein the portfolio data is from heterogeneous data sources that store data in a plurality of formats.

25. The system of claim 23 wherein the portfolio data represents investments from distinctive parts of an organization.

26. The system of claim 23 wherein the organization parts include at least two of management, marketing, sales, product management, manufacturing, research, development, IT, finance, operations, consulting, engineering, and human resources.

27. The system of claim 23 wherein the portfolio data includes financial investments and project management characterizations.

28. The system of claim 23 wherein the plurality of views includes at least one dimensioned attribute.

29. The system of claim 23 wherein the dimensioned attribute includes at least two of actual value, baseline value, plan value, target value, and scenario value.

30. The system of claim 23, further comprising:

a meta-object data management system for managing the portfolio representation structure.

31. The system of claim 30 wherein the structure is hierarchical.

32. A method in a computer system for representing, managing, and analyzing investments of an organization, comprising:

instantiating a hierarchy of object instances, each object instance representing an investment of the organization, at least two object instances representing data from different categories of investments;

receiving a request to display data from a plurality of the object instances according to an attribute specification; and

displaying the object instances of the plurality of object instances that match the attribute specification, in a manner that is in accordance with the attribute specification, so that multi-dimensional views of the matching objects are computed and displayed dynamically.

33. The method of claim 32, further comprising supporting changes to the hierarchy of object instances while displaying the object instances that match the attribute specification, thereby concurrently allowing online transaction processing while supporting online analysis.

34. The method of claim 32 wherein the request to display data according to an attribute specification is received as properties of a datasheet associated with an object instance in the hierarchy.

35. The method of claim 32, each object instance having constituent attributes with associated values, wherein the displaying of the object instances that match the attribute specification further comprises:

determining from each matching object instance the constituent attributes and associated values;

dynamically constructing a virtual object instance to represent each aggregation of matching objects that is not already an instantiated object; and

displaying the constructed virtual objects along with the matching object instances.

36. The method of claim 35 wherein each virtual object instance automatically aggregates appropriate attributes of children object instances and children virtual object instances.

37. The method of claim 36 wherein at least one of the aggregated attributes represents time-phased data values.

38. The method of claim 36 wherein the attributes of the children instances specify whether they are to be aggregated to a parent virtual object instance.

39. The method of claim 35 wherein a value associated with at least one attribute is a dimensioned value.

40. The method of claim 39 wherein the dimensioned value comprises at least two of actual value, baseline value, plan value, target value, and scenario value.

41. The method of claim 32 wherein the datasheet defines a dynamically created multi-dimensional view of the instantiated object hierarchy that is not previously stored in the computer system.

42. The method of claim 32 wherein each child object instance of a parent object instance represents a relationship whereby the investment associated with the parent object instance is an aggregation of the child object instances associated with the parent object instance.

43. The method of claim 32 wherein the investments are part of an enterprise portfolio management system.

44. The method of claim 43 wherein the investments comprise heterogeneous data types in the form of at least two of financial investments, project management, collections of projects, programs, assets, human resources, products, portfolios, initiatives, applications, operations, processes, and activities.

45. The method of claim 43, the enterprise portfolio management system managing enterprise portfolios, wherein the enterprise portfolios comprise at least two of engineering, marketing, product management, manufacturing, sales, information technology, finance, human resources, research, development, and professional services portfolios.

46. The method of claim 32 wherein the investments include time-phased data.

47. The method of claim 46 wherein the time-phased data is stored in a manner that indicates a first designated time period in a virtual object instance and is stored in a manner that indicates a second designated time period in the object instances that are children of the virtual object instance that stores the time-phased data.

48. The method of claim 47 wherein the first designated time period is at least one of weekly, monthly, quarterly, and annually.

49. The method of claim 47 wherein the second designated time period is at least one of daily and hourly.

50. The method of claim 47 wherein the second designated time period is a custom defined range.

51. The method of claim 47 wherein the first designated time period is a custom defined range.

52. The method of claim 46 wherein a change to time-phased data of a virtual object instance is automatically reflected in changes to object instances that are children of the virtual object instance in which the change occurred.

53. The method of claim 32 wherein the object instances of the plurality that match the attribute specification is filtered based upon a security role associated with the request.

54. The method of claim 32, further comprising:

indicating that the request to display data according to the attribute specification is to be applied to a different plurality of object instances of the hierarchy; and

automatically modifying the display of the object instances to include object instances from the different plurality that match the attribute specification.

55. The method of claim 54 wherein the indication is to copy the request and the display of object instances includes the object instances from the

plurality that match the attribute specification in addition to objects instances from the different plurality.

56. The method of claim 54 wherein the indication is to move the request and the object instances from the plurality that match the attribute specification are removed from the display unless the object instances from the plurality that match are also instances of the different plurality.

57. The method of claim 32, each object instance having constituent attributes with associated values, further comprising:

associating a chart definition with the request to display data according to the attribute specification, the chart definition having an associated set of chart parameters; and

automatically determining values for the associated chart parameters based upon the constituent attributes of the object instances that match the attribute specification.

58. The method of claim 57, further comprising:

presenting a chart that corresponds to the chart definition, such that the values displayed by the chart are automatically determined from the values associated with the constituent attributes of the object instances that match the attribute specification.

59. The method of claim 57, further comprising:

associating the chart definition with a request to display data from a different plurality of object instances according to a second attribute specification; and

automatically re-determining the values for the associated chart parameters based upon the constituent attributes of the object instances from the different plurality that match the second attribute specification.

60. The method of claim 59 wherein the second attribute specification is the same as the attribute specification.

61. The method of claim 59 wherein the chart definition is moved.

62. The method of claim 59 wherein the chart definition is copied.

63. A computer-readable memory medium containing instructions for controlling a computer system to represent, manage, and analyze investments of an organization, by:

instantiating a hierarchy of object instances, each object instance representing an investment of the organization, at least two object instances representing data from different types of investments;

receiving a request to display data from a plurality of the object instances according to an attribute specification; and

displaying the object instances of the plurality of object instances that match the attribute specification, in a manner that is in accordance with the attribute specification, so that multi-dimensional views of the matching objects are computed and displayed dynamically.

64. The memory medium of claim 63, further comprising instructions that control the computer processor by concurrently allowing online transaction processing while supporting online analysis by supporting changes to the hierarchy of object instances while displaying the object instances that match the attribute specification.

65. The memory medium of claim 63 wherein the request to display data according to an attribute specification is received as properties of a datasheet associated with an object instance in the hierarchy.

66. The memory medium of claim 63, each object instance having constituent attributes with associated values, wherein the displaying of the object instances that match the attribute specification further comprises:

determining from each matching object instance the constituent attributes and associated values;

dynamically constructing a virtual object instance to represent each aggregation of matching objects that is not already an instantiated object; and

displaying the constructed virtual objects along with the matching object instances.

67. The memory medium of claim 66 wherein each virtual object instance automatically aggregates appropriate attributes of children object instances and children virtual object instances.

68. The memory medium of claim 67 wherein at least one of the aggregated attributes represents time-phased data values.

69. The memory medium of claim 66 wherein a value associated with at least one attribute is a dimensioned value.

70. The memory medium of claim 69 wherein the dimensioned value comprises at least two of actual value, baseline value, plan value, target value, and scenario value.

71. The memory medium of claim 63 wherein the datasheet defines a dynamically created multi-dimensional view of the instantiated object hierarchy that is not previously stored in the computer system.

72. The memory medium of claim 63 wherein each child object instance of a parent object instance represents a relationship whereby the investment associated with the parent object instance is an aggregation of the child object instances associated with the parent object instance.

73. The memory medium of claim 63 wherein the investments are part of an enterprise portfolio management system.

74. The memory medium of claim 63 wherein the investments include time-phased data.

75. The memory medium of claim 74 wherein the time-phased data is stored in a manner that indicates a first designated time period in a virtual object instance and is stored in a manner that indicates a second designated time period in the object instances that are children of the virtual object instance that stores the time-phased data.

76. The memory medium of claim 74 wherein a change to time-phased data of a virtual object instance is automatically reflected in changes to object instances that are children of the virtual object instance in which the change occurred.

77. The memory medium of claim 63 wherein the object instances of the plurality that match the attribute specification is filtered based upon a security role associated with the request.

78. The memory medium of claim 63, further comprising instructions that control the computer processor by:

indicating that the request to display data according to the attribute specification is to be applied to a different plurality of object instances of the hierarchy; and

automatically modifying the display of the object instances to include object instances from the different plurality that match the attribute specification.

79. The memory medium of claim 63, each object instance having constituent attributes with associated values, further comprising instructions that control the computer processor by:

associating a chart definition with the request to display data according to the attribute specification, the chart definition having an associated set of chart parameters; and

automatically determining values for the associated chart parameters based upon the constituent attributes of the object instances that match the attribute specification.

80. The memory medium of claim 79, further comprising instructions that control the computer processor by

presenting a chart that corresponds to the chart definition, such that the values displayed by the chart are automatically determined from the values associated with the constituent attributes of the object instances that match the attribute specification.

81. The memory medium of claim 79, further comprising instructions that control the computer processor by:

associating the chart definition with a request to display data from a different plurality of object instances according to a second attribute specification; and automatically re-determining the values for the associated chart parameters based upon the constituent attributes of the object instances from the different plurality that match the second attribute specification.

82. A portfolio management system for representing, managing, and analyzing investments of an organization, comprising:

a portfolio management component that is structured to instantiate a hierarchy of object instances according to received data, each object instance representing an investment of the organization, at least two object instances representing data from different types of investments; and

a portfolio analysis component that is structured to

receive a request to display data from a plurality of the object instances according to an attribute specification; and

display the object instances of the of the plurality of object instances that match the attribute specification in a manner that is in accordance with the attribute specification, so that multi-dimensional views of the matching objects are computed and displayed dynamically.

83. The system of claim 82 wherein the system concurrently allows online transaction processing while supporting online analysis by supporting changes to the hierarchy of object instances while displaying the object instances that match the attribute specification.

84. The system of claim 82 wherein the request to display data according to an attribute specification is received as properties of a datasheet associated with an object instance in the hierarchy.

85. The system of claim 82, each object instance having constituent attributes with associated values, wherein the portfolio analysis component is further structured to display of the object instances that match the attribute specification by:

determining from each matching object instance the constituent attributes and associated values;

dynamically constructing a virtual object instance to represent each aggregation of matching objects that is not already an instantiated object; and

displaying the constructed virtual objects along with the matching object instances.

86. The system of claim 85 wherein each virtual object instance automatically aggregates appropriate attributes of children object instances and children virtual object instances.

87. The system of claim 86 wherein at least one of the aggregated attributes represents time-phased data values.

88. The system of claim 85 wherein a value associated with at least one attribute is a dimensioned value.

89. The system of claim 88 wherein the dimensioned value comprises at least two of actual value, baseline value, plan value, target value, and scenario value.

90. The system of claim 82 wherein the datasheet defines a dynamically created multi-dimensional view of the instantiated object hierarchy that is not previously stored in the computer system.

91. The system of claim 82 wherein each child object instance of a parent object instance represents a relationship whereby the investment associated with

the parent object instance is an aggregation of the child object instances associated with the parent object instance.

92. The system of claim 82 wherein the investments are part of an enterprise portfolio management system.

93. The system of claim 82 wherein the investments include time-phased data.

94. The system of claim 93 wherein the time-phased data is stored in a manner that indicates a first designated time period in a virtual object instance and is stored in a manner that indicates a second designated time period in the object instances that are children of the virtual object instance that stores the time-phased data.

95. The system of claim 93 wherein a change to time-phased data of a virtual object instance is automatically reflected in changes to object instances that are children of the virtual object instance in which the change occurred.

96. The system of claim 82 wherein the object instances of the plurality that match the attribute specification is filtered based upon a security role associated with the request.

97. The system of claim 82 wherein the portfolio analysis component is further structured to:

receive an indication that the request to display data according to the attribute specification is to be applied to a different plurality of object instances of the hierarchy; and automatically modify the display of the object instances to include object instances from the different plurality that match the attribute specification.

98. The system of claim 82, each object instance having constituent attributes with associated values, further comprising:

a charting tool that is structured to:

associate a chart definition with the request to display data according to the attribute specification, the chart definition having an associated set of chart parameters; and

automatically determine values for the associated chart parameters based upon the constituent attributes of the object instances that match the attribute specification.

99. The system of claim 98 wherein the charting tool is further structured to present a chart that corresponds to the chart definition, such that the values displayed by the chart are automatically determined from the values associated with the constituent attributes of the object instances that match the attribute specification.

100. The system of claim 98 wherein the charting tool is further structured to:

associate the chart definition with a request to display data from a different plurality of object instances according to a second attribute specification; and

automatically re-determine the values for the associated chart parameters based upon the constituent attributes of the object instances from the different plurality that match the second attribute specification.

101. A method in a computer system for normalizing the representation of diverse data entities, each data entity being associated with an entity type that

defines a plurality of associated attributes, and each data entity having an associated plurality of attribute values that correspond to the plurality of attributes associated with the entity type, comprising:

instantiating a meta-object to represent one of the diverse data entities;

storing a type indicator in the meta-object to indicate a first entity type; and

storing values for a plurality of attributes of the meta-object in a single data variable, such that the plurality of attributes of the meta-object are determined based upon the entity type indicated by the stored type indicator and the values that are stored are based upon the plurality of attribute values associated with the represented data entity.

102. The method of claim 101, further comprising:

modifying the type indicator in the meta-object to a second indicator that indicates a second entity type; and

automatically adjusting the plurality of attributes of the meta-object to correspond to the second entity type.

103. The method of claim 102, the automatic adjusting further comprising:

eliminating stored values for attributes that are not associated with the second entity type; and

automatically determining and storing values for attributes that are associated with the second entity type but not associated with the first entity type.

104. The method of claim 102, the meta-object instantiated as part of a hierarchy of instantiated meta-objects and having an associated position in the hierarchy, wherein at least one attribute of the plurality of attributes of the meta-object is

automatically adjusted by determining a value based upon the associated position of the meta-object in the hierarchy.

105. The method of claim 104 wherein the value of the adjusted attribute is determined by aggregating the stored values of corresponding attributes in meta-objects that are children of the meta-object in the hierarchy.

106. The method of claim 101 wherein the definition of plurality of attributes associated with the first entity type is modified, and wherein the stored values for the plurality of attributes of the meta object are automatically updated to reflect the modification.

107. The method of claim 101, the meta-object instantiated as part of a hierarchy of instantiated meta-objects and having an associated position in the hierarchy, wherein at least one stored value of the plurality of attributes of the meta-object is automatically determined based upon the associated position of the meta-object in the hierarchy.

108. The method of claim 107 wherein the at least one stored value of the plurality of attributes is determined by aggregating the stored values of corresponding attributes in meta-objects that are children of the meta-object in the hierarchy.

109. The method of claim 107 wherein a meta-object in the hierarchy can be positioned as a child of any other meta-object in the hierarchy, regardless of the entity type indicated by the stored type indicator of the meta-object to be positioned.

110. The method of claim 101 wherein the single data variable contains a plurality of tags, each of which indicates an identity of an attribute and an associated attribute value.

111. The method of claim 110 wherein the tags are XML tag pairs, each tag pair identifying an attribute and delimiting a value associated with the identified attribute.

112. The method of claim 110 wherein the single data variable contents are stored in a data repository on a persistent physical storage device.

113. The method of claim 101 wherein the single data variable contains a serialized representation of attribute name-value pairs associated with the meta-object.

114. The method of claim 101 wherein at least one of the diverse data entities corresponds to data associated with a financial investment and at least one of the diverse data entities corresponds to data associated with a project management investment.

115. The method of claim 101 wherein the diverse data entities correspond to at least two of a financial investment, project, asset, product, program, resource, initiative, application, operation, process, and activity.

116. The method of claim 101 as used to create canonical representations of data stored in heterogeneous data repositories.

117. The method of claim 101 in an enterprise portfolio management system.

118. The method of claim 117 wherein the diverse entities comprise portfolios, projects, products, collections of projects, assets, resources, investments, programs, initiatives, applications, operations, processes, and activities.

119. The method of claim 117 wherein the diverse entities comprise data from different organizations within the enterprise.

120. The method of claim 117, the enterprise portfolio management system managing enterprise portfolios, wherein the enterprise portfolios comprise at least two of engineering, marketing, product management, manufacturing, sales, information technology, finance, product, human resources, research, development, and professional service portfolios and the diverse entities are contained in the at least two portfolios.

121. The method of claim 101 wherein a stored value of at least one of the plurality of attributes of the meta-object includes time-phased data.

122. The method of claim 121 wherein the time-phased data includes a value, a start time indication, and a duration.

123. The method of claim 121 wherein the time-phased data includes a plurality of value, start time indication, and duration triplets.

124. The method of claims 123 wherein the duration specifies a range of time.

125. The method of claims 123 wherein the time-phased data indicates values that correspond to at least one of hourly, daily, quarterly, monthly, yearly, and semi-annual time periods.

126. The method of claim 101 wherein meta-objects are instantiated in a hierarchy based upon security roles.

127. The method of claim 101 wherein a stored value of at least one of the plurality of attributes of the meta-object is a dimensioned value.

128. The method of claim 127 wherein the dimensioned value comprises at least two of actual value, baseline value, plan value, target value, and scenario value.

129. A computer-readable memory medium containing instructions for controlling a computer processor to normalize the representation of diverse data entities, each data entity being associated with a type that defines a plurality of attributes that are associated with the type of data entity, and each entity having a plurality of attribute values that correspond to the plurality of attributes associated with the type of data entity, by:

instantiating a meta-object to represent one of the diverse data entities;

storing a type indicator in the meta-object to indicate a first entity type; and

storing values for a plurality of attributes of the meta-object in a single data variable, such that the plurality of attributes of the meta-object are determined based upon the entity type indicated by the stored type indicator and the values that are stored are based upon the plurality of attribute values associated with the represented data entity.

130. The memory medium of claim 129, further comprising instructions that control the computer processor by:

modifying the type indicator in the meta-object to a second indicator that indicates a second entity type; and automatically adjusting the plurality of attributes of the meta-object to correspond to the second entity type.

131. The memory medium of claim 130, the automatic adjusting further comprising:

eliminating stored values for attributes that are not associated with the second entity type; and

automatically determining and storing values for attributes that are associated with the second entity type but not associated with the first entity type.

132. The memory medium of claim 130, the meta-object instantiated as part of a hierarchy of instantiated meta-objects and having an associated position in the hierarchy, wherein at least one attribute of the plurality of attributes of the meta-object is automatically adjusted by determining a value based upon the associated position of the meta-object in the hierarchy.

133. The memory medium of claim 132 wherein the value of the adjusted attribute is determined by aggregating the stored values of corresponding attributes in meta-objects that are children of the meta-object in the hierarchy.

134. The memory medium of claim 129 wherein the definition of plurality of attributes associated with the first entity type is modified, and wherein the stored values for the plurality of attributes of the meta object are automatically updated to reflect the modification.

135. The memory medium of claim 129, the meta-object instantiated as part of a hierarchy of instantiated meta-objects and having an associated position in the hierarchy, wherein at least one stored value of the plurality of attributes of the meta-

object is automatically determined based upon the associated position of the meta-object in the hierarchy.

136. The memory medium of claim 135 wherein the at least one stored value of the plurality of attributes is determined by aggregating the stored values of corresponding attributes in meta-objects that are children of the meta-object in the hierarchy.

137. The memory medium of claim 135 wherein a meta-object in the hierarchy can be positioned as a child of any other meta-object in the hierarchy, regardless of the entity type indicated by the stored type indicator of the meta-object to be positioned.

138. The memory medium of claim 129 wherein the single data variable contains a plurality of tags, each of which indicates an identity of an attribute and an associated attribute value.

139. The memory medium of claim 138 wherein the tags are XML tag pairs, each tag pair identifying an attribute and delimiting a value associated with the identified attribute.

140. The memory medium of claim 138 wherein the single data variable contents are stored in a data repository on a persistent physical storage device.

141. The memory medium of claim 129 wherein the single data variable contains a serialized representation of attribute name-value pairs associated with the meta-object.

142. The memory medium of claim 129 wherein at least one of the diverse data entities corresponds to data associated with a financial investment and at least one of the diverse data entities corresponds to data associated with a project management investment.

143. The memory medium of claim 129 wherein the diverse data entities correspond to at least two of a financial investment, project, asset, product, program, resource, initiative, application, operation, process, and activity.

144. The memory medium of claim 129 as used to create canonical representations of data stored in heterogeneous data repositories.

145. The memory medium of claim 129 in an enterprise portfolio management system.

146. The memory medium of claim 145 wherein the diverse entities comprise portfolios, projects, products, collections of projects, assets, resources, investments, programs, initiatives, applications, operations, processes, and activities.

147. The memory medium of claim 129 wherein a stored value of at least one of the plurality of attributes of the meta-object includes time-phased data.

148. The memory medium of claim 147 wherein the time-phased data includes a value, a start time indication, and a duration.

149. The memory medium of claim 129 wherein meta-objects are instantiated in a hierarchy based upon security roles.

150. The memory medium of claim 129 wherein a stored value of at least one of the plurality of attributes of the meta-object is a dimensioned value.

151. The memory medium of claim 150 wherein the dimensioned value comprises at least two of actual value, baseline value, plan value, target value, and scenario value.

152. An object type management system for canonically representing diverse data entities, each having different collections of attributes, comprising:

a plurality of object type definitions, each definition associated with a plurality of attributes and corresponding to a one of the diverse data entities; and

an instantiation mechanism that is structured to

instantiate a meta-object for a designated data entity;

store an indication of the object type definition that is associated with the designated data entity;

storing, in a single field of the meta object, for each of the attributes associated with the indicated object type definition, an indication of the associated attribute and a value that is based upon the designated data entity.

153. The system of claim 152, the instantiation mechanism further structured to:

modify the indication of the object type definition stored in the meta-object to a second indication that indicates a second entity type; and

automatically adjust the plurality of attributes stored in the single field of the meta-object to correspond to the attributes associated with the second entity type.

154. The system of claim 153, the automatic adjustment further comprising:

eliminating stored values for attributes that are not associated with the second entity type; and

automatically determining and storing values for attributes that are associated with the second entity type but not associated with the first entity type.

155. The system of claim 153, the meta-object instantiated as part of a hierarchy of instantiated meta-objects and having an associated position in the hierarchy, wherein at least one attribute of the plurality of attributes of the meta-object is automatically adjusted by determining a value based upon the associated position of the meta-object in the hierarchy.

156. The system of claim 155 wherein the value of the adjusted attribute is determined by aggregating the stored values of corresponding attributes in meta-objects that are children of the meta-object in the hierarchy.

157. The system of claim 152 wherein the plurality of attributes associated with the first entity type is modified, and wherein the stored values for the plurality of attributes of the meta object are automatically updated to reflect the modification.

158. The system of claim 152, the meta-object instantiated as part of a hierarchy of instantiated meta-objects and having an associated position in the hierarchy, wherein at least one stored value of the plurality of attributes of the meta-object is automatically determined based upon the associated position of the meta-object in the hierarchy.

159. The system of claim 158 wherein the at least one stored value of the plurality of attributes is determined by aggregating the stored values of corresponding attributes in meta-objects that are children of the meta-object in the hierarchy.

160. The system of claim 158 wherein a meta-object in the hierarchy can be positioned as a child of any other meta-object in the hierarchy, regardless of the entity type associated with the object type definition indication that is stored in the meta-object to be positioned.

161. The system of claim 152 wherein the single field contains a plurality of tags, each of which indicates an identity of an attribute and an associated attribute value.

162. The system of claim 161 wherein the tags are XML tag pairs, each tag pair identifying an attribute and delimiting a value associated with the identified attribute.

163. The system of claim 161 wherein the single field contents are stored in a data repository on a persistent physical storage device.

164. The system of claim 152 wherein the single field contains a serialized representation of attribute name-value pairs associated with the meta-object.

165. The system of claim 152 wherein at least one of the diverse data entities corresponds to data associated with a financial investment and at least one of the diverse data entities corresponds to data associated with a project management investment.

166. The system of claim 152 wherein the diverse data entities correspond to at least two of a financial investment, project, asset, product, program, resource, initiative, application, operation, process, and activity.

167. The system of claim 152 as used to create canonical representations of data stored in heterogeneous data repositories.

168. The system of claim 152 used in an enterprise portfolio management system.

169. The system of claim 152 wherein a stored value of at least one of the plurality of attributes stored in the meta-object includes time-phased data.

170. The system of claim 169 wherein the time-phased data includes a value, a start time indication, and a duration.

171. The system of claim 152 wherein meta-objects are instantiated in a hierarchy based upon security roles.

172. The system of claim 152 wherein a stored value of at least one of the plurality of attributes of the meta-object is a dimensioned value.

173. A method in a computer system for creating a dynamic hierarchy of objects, wherein the hierarchy is defined by a containment structure of instantiated objects instead of object class definitions, comprising:

defining a plurality of object classes, each class defining a plurality of attributes;

instantiating a first object of a first class, the first object having a set of attributes as defined by the first class;

instantiating a second object of a second class as a child object of the first object in the hierarchy, the second object having a second set of attributes as defined by the second class, wherein the values of the set of attributes of the second object are used to calculate the values of the set of attributes of the first object; and

instantiating a third object of the first class as a child object of the second object in the hierarchy, the third object having a set of attributes as defined by the first class, wherein the values of the set of attributes of the third object are used to calculate the values of the set of attributes of the second object.

174. The method of claim 173 wherein the first and third objects relate to financial data and the second object relates to project management data.

175. The method of claim 173 wherein at least one of the objects relate to project management data and at least one of the objects relates to financial data.

176. The method of claim 173 wherein at least one of the attributes of the set of attributes of the second object that is used to calculate a value of an attribute of the first object is a different type of attribute than the attribute of the first object whose value is calculated.

177. The method of claim 176 wherein the attribute whose value is calculated is an expense attribute and the attribute used to calculate the value is not an expense attribute.

178. The method of claim 176 wherein the attribute whose value is calculated is a resource attribute and the attribute used to calculate the value is not a resource attribute.

179. The method of claim 173 wherein the objects represent investments.

180. The method of claim 179 wherein the investments represent at least one of real property, commercial paper, and equity investments.

181. The method of claim 179 wherein the investments represent at least one of projects, products, expenses, resources, assets, programs, portfolios, initiatives, applications, operations, processes, and activities.

182. The method of claim 179 wherein the investments represent at least one of financial, marketing, sales, manufacturing, human resources, information technology, services, engineering, research, development, consulting, product research, product management, and product development data.

183. The method of claim 173 wherein the hierarchy represents an enterprise portfolio management system.

184. The method of claim 173 wherein an object is instantiated as a child object of a parent object when appropriate according to a security role associated with the parent object.

185. The method of claim 173 wherein at least one of the attributes of at least one of the objects is a dimensioned attribute.

186. The method of claim 185 wherein the dimensioned attribute comprises at least two of actual value, baseline value, plan value, target value, and scenario value.

187. A computer-readable memory medium containing instructions for controlling a computer processor to create a dynamic hierarchy of objects, wherein the hierarchy is defined by a containment structuring of instantiated objects instead of object class definitions, by:

defining a plurality of object classes, each class defining a plurality of attributes;

instantiating a first object of a first class, the first object having a set of attributes as defined by the first class;

instantiating a second object of a second class as a child object of the first object in the hierarchy, the second object having a second set of attributes as defined by the second class, wherein the values of the set of attributes of the second object are used to calculate the values of the set of attributes of the first object; and

instantiating a third object of the first class as a child object of the second object in the hierarchy, the third object having a set of attributes as defined by the first class, wherein the values of the set of attributes of the third object are used to calculate the values of the set of attributes of the second object.

188. The memory medium of claim 187 wherein the first and third objects relate to financial data and the second object relates to project management data.

189. The memory medium of claim 187 wherein at least one of the objects relate to project management data and at least one of the objects relates to financial data.

190. The memory medium of claim 187 wherein at least one of the attributes of the set of attributes of the second object that is used to calculate a value of an attribute of the first object is a different type of attribute than the attribute of the first object whose value is calculated.

191. The memory medium of claim 190 wherein the attribute whose value is calculated is an expense attribute and the attribute used to calculate the value is not an expense attribute.

192. The memory medium of claim 190 wherein the attribute whose value is calculated is a resource attribute and the attribute used to calculate the value is not a resource attribute.

193. The memory medium of claim 187 wherein the objects represent investments.

194. The memory medium of claim 193 wherein the investments represent at least one of real property, commercial paper, and equity investments.

195. The memory medium of claim 193 wherein the investments represent at least one of projects, products, expenses, resources, assets, programs, portfolios, initiatives, applications, operations, processes, and activities.

196. The memory medium of claim 193 wherein the investments represent at least one of financial, marketing, sales, manufacturing, human resources, information technology, services, engineering, research, development, consulting, product research, product management, and product development data.

197. The memory medium of claim 187 wherein the hierarchy represents an enterprise portfolio management system.

198. The memory medium of claim 187 wherein an object is instantiated as a child object of a parent object when appropriate according to a security role associated with the parent object.

199. The memory medium of claim 187 wherein at least one of the attributes of at least one of the objects is a dimensioned attribute.

200. The memory medium of claim 187 wherein the dimensioned attribute comprises at least two of actual value, baseline value, plan value, target value, and scenario value.

201. A computer system for creating a dynamic hierarchy of objects, wherein the hierarchy is defined by the containment of instantiated objects instead of object class definitions, comprising:

a plurality of object class definitions including a first class and a second class, each class defining a plurality of attributes; and

an instantiation mechanism that is structured to

instantiate a first object of a first class, the first object having a set of attributes as defined by the first class;

instantiate a second object of a second class as a child object of the first object in the hierarchy, the second object having a second set of attributes as defined by the second class, wherein the values of the set of attributes of the second object are used to calculate the values of the set of attributes of the first object; and

instantiate a third object of the first class as a child object of the second object in the hierarchy, the third object having a set of attributes as defined by the first class, wherein the values of the set of attributes of the third object are used to calculate the values of the set of attributes of the second object.

202. The system of claim 201 used as an enterprise portfolio management system.

203. The system of claim 201 wherein the first and third objects relate to financial data and the second object relates to project management data.

204. The system of claim 201 wherein at least one of the objects relate to project management data and at least one of the objects relates to financial data.

205. The system of claim 201 wherein at least one of the attributes of the set of attributes of the second object that is used to calculate a value of an attribute of the first object is a different type of attribute than the attribute of the first object whose value is calculated.

206. The system of claim 205 wherein the attribute whose value is calculated is an expense attribute and the attribute used to calculate the value is not an expense attribute.

207. The system of claim 205 wherein the attribute whose value is calculated is a resource attribute and the attribute used to calculate the value is not a resource attribute.

208. The system of claim 201 wherein the objects represent investments.

209. The system of claim 208 wherein the investments represent at least one of real property, commercial paper, and equity investments.

210. The system of claim 208 wherein the investments represent at least one of projects, products, expenses, resources, assets, programs, portfolios, initiatives, applications, operations, processes, and activities.

211. The system of claim 208 wherein the investments represent at least one of financial, marketing, sales, manufacturing, human resources, information

technology, services, engineering, research, development, consulting, product research, product management, and product development data.

212. The system of claim 201 wherein an object is instantiated as a child object of a parent object when appropriate according to a security role associated with the parent object.

213. The system of claim 201 wherein at least one of the attributes of at least one of the objects is a dimensioned attribute.

214. The system of claim 213 wherein the dimensioned attribute comprises at least two of actual value, baseline value, plan value, target value, and scenario value.

215. A method in a computer system for creating a storage repository of serialized data using tags, comprising:

allocating a portion of memory to store a plurality of attribute name and associated value pairs; and

for each pair of the plurality of attribute and associated value pairs, creating a tagged attribute pair by,

storing in the allocated memory a beginning tag that indicates the attribute name of the pair;

storing, after the beginning tag, the value associated with the pair; and

storing after the value, an ending tag that indicates that there is no more data associated with the pair.

216. The method of claim 215, further comprising:

receiving a designated attribute name; and determining a corresponding value for the received attribute name by

retrieving from the portion of memory a beginning tag that corresponds to the attribute name; and

retrieving the value stored after the retrieved beginning tag.

217. The method of claim 215 wherein the stored attribute name and associated value pairs represent state information for an instantiated object.

218. The method of claim 217 wherein the object is an investment in a portfolio management system.

219. The method of claim 217 wherein the portion of memory is stored on a persistent data repository device.

220. The method of claim 217, the instantiated object having an associated type that defines the attribute name and associated value pairs that are stored, further comprising:

changing the type associated with the instantiated object to a new type by:

determining the attribute name and associated value pairs defined by the new type;

determining a set of tagged attribute pairs from the storage repository that are not included in the attribute name and associated value pairs defined by the new type and causing the determined set of tagged attribute pairs to be no longer accessible; and

storing new tagged attributes in the storage repository for attribute name and associated value pairs that are defined by the new type and are not also defined by the type associated with the instantiated object.

221. The method of claim 217 wherein the storage repository is stored as a single variable in the instantiated object.

222. The method of claim 217 wherein the storage repository is used to normalize a plurality of types of objects into a canonical form.

223. The method of claim 215 wherein the tags are XML tags.

224. The method of claim 215 wherein at least one value stored between a beginning tag and an ending tag stores time-phased data.

225. The method of claim 224 wherein the time-phased data is stored as collections that specify a value, a start time indicator, and a duration.

226. The method of claim 224 wherein the time-phased data can be converted to units other than the units of the stored time-phased data.

227. The method of claim 224 wherein the time-phased data includes daily as well as cumulative time period information.

228. A data processing machine comprising a computer processor and a memory, the memory containing a representation of structured data that is manipulated by the computer processor according to a computer program, the structured data being stored in serialized form in the memory as:

a collection of tagged attribute name and associated value pairs, each tagged pair being stored as:

a beginning tag that indicates the name of the attribute;
the associated value; and
an ending tag that indicates the end of the tagged pair.

229. The data processing machine of claim 228 wherein the tags are XML tags.

230. The data processing machine of claim 228 wherein the tagged pairs represent state information for an instantiated object.

231. The data processing machine of claim 230 wherein the object is an investment in a portfolio management system.

232. The data processing machine of claim 228 wherein the memory is stored on a persistent data repository device.

233. The data processing machine of claim 228 wherein at least one of the stored values is data that represents a time-phased attribute.

234. The data processing machine of claim 233 wherein the time-phased data is stored as collections that specify a value, a start time indicator, and a duration.

235. The data processing machine of claim 233 wherein the time-phased data can be converted to units other than the units of the stored time-phased data.

236. The data processing machine of claim 233 wherein the time-phased data includes daily as well as cumulative time period information.